CLAIMS:

We claim:

1. A method for selecting a cache memory allocation to provide an optimized target cache hit rate in a caching component of a content delivery system, the method comprising the steps of:

identifying a current cache size and a contemporaneously experienced trace footprint;

determining a hit rate produced in response to said current cache size and said contemporaneously experienced trace footprint;

computing a Zipf alpha coefficient for said current cache size, trace footprint and hit rate;

selecting an optimal hit rate; and,

further computing an optimal cache size for said Zipf alpha coefficient, trace footprint and optimal hit rate.

2. The method of claim 1, further comprising the step of reconfiguring the cache memory allocation based upon said optimal cache size.

3. The method of claim 1, wherein said computing step comprises the step of computing said Zipf alpha coefficient based upon the equation

$$HitRate = \frac{1 - m^{1-\alpha}}{1 - T^{1-\alpha}}$$

where α is said Zipf alpha coefficient, m the known size of the cache, T is said trace footprint and HitRate is a contemporaneously experienced hit rate.

4. The method of claim 1, wherein said determining step comprises parsing a log of server activity to identify said hit rate.

5. The method of claim 1, wherein said identifying step comprises the step of identifying a current cache size and a contemporaneously experienced trace footprint for a cluster of servers.

6. The method of claim 1, wherein said identifying step comprises the step of identifying a current cache size and a contemporaneously experienced trace footprint for a single content delivery server.

7. A system for selecting a cache memory allocation to provide an optimized target cache hit rate in a caching component of a content delivery system, the system comprising a Zipf alpha coefficient parameter computation processor coupled to an optimal cache size computation processor communicatively linked to a cache in the content delivery system.

8. The system of claim 7, further comprising a communicative linkage between said Zipf alpha parameter computation processor and a server log storing statistics related to a hit rate for said cache over which said hit rate can be communicated to said Zipf alpha parameter computation processor.

9. The system of claim 7, wherein said Zipf alpha parameter computation processor comprises programming for computing a Zipf alpha coefficient for a supplied hit rate, cache size and trace footprint according to the equation

$$HitRate = \frac{1 - m^{1-\alpha}}{1 - T^{1-\alpha}}$$

where α is said Zipf alpha coefficient, m is said cache size, T is said trace footprint and HitRate is said hit rate.

10. The system of claim 7, wherein said optimal cache size computation processor comprises programming for computing an optimal cache size for a supplied Zipf alpha coefficient, a preferred hit rate, and a known trace footprint according to the equation

$$HitRate = \frac{1 - m^{1-\alpha}}{1 - T^{1-\alpha}}$$

where α is said Zipf alpha coefficient, m is said optimal cache size, T is said known trace footprint and HitRate is said preferred hit rate.

11. A machine readable storage having stored thereon a computer program for selecting a cache memory allocation to provide an optimized target cache hit rate in a caching component of a content delivery system, the computer program comprising a routine set of instructions which when executed cause the machine to perform the steps of:

identifying a current cache size and a contemporaneously experienced trace footprint;

determining a hit rate produced in response to said current cache size and said contemporaneously experienced trace footprint;

computing a Zipf alpha coefficient for said current cache size, trace footprint and hit rate;

selecting an optimal hit rate; and,

further computing an optimal cache size for said Zipf alpha coefficient, trace footprint and optimal hit rate.

12. The machine readable storage of claim 11, further comprising the step of reconfiguring the cache memory allocation based upon said optimal cache size.

13. The machine readable storage of claim 11, wherein said computing step comprises the step of computing said Zipf alpha coefficient based upon the equation

$$HitRate = \frac{1 - m^{1-\alpha}}{1 - T^{1-\alpha}}$$

where α is said Zipf alpha coefficient, m the known size of the cache, T is said trace footprint and HitRate is a contemporaneously experienced hit rate.

14. The machine readable storage of claim 11, wherein said determining step comprises parsing a log of server activity to identify said hit rate.

15. The machine readable storage of claim 11, wherein said identifying step comprises the step of identifying a current cache size and a contemporaneously experienced trace footprint for a cluster of servers.

16. The machine readable storage of claim 11, wherein said identifying step comprises the step of identifying a current cache size and a contemporaneously experienced trace footprint for a single content delivery server.